

Projet Base de Données

COMPTE-RENDU

Julie SAOULI

Maxime GOURGOULHON

Pierre BOUVIER

Geoffrey TANGUY

Jérôme THÉATE

ENSIMAG | Enseignant encadrant : M. Christophe BOBINEAU | 2017 - 2018

Table des matières

INTRODUCTION

L'agence de voyage AWHY, forte de son succès, entreprend sa transformation numérique.

Elle tient à numériser ses outils de gestion tout en conservant une étroite collaboration avec ses partenaires.

De ce fait, certaines règles d'implémentation devront être respectées afin d'assurer le fonctionnement des outils numériques et la compatibilité des bases de données communes entre tous les partenaires.

Dans le cadre du développement de la société AWHY, nous avons été mandaté afin de réaliser la conception et l'implémentation de leur base de données. Ce rapport vous apportera les outils nécessaires à la compréhension de la conception de cette base de données. Et vous permettra en outre de prendre connaissance des enjeux et des difficultés qui ont permis de privilégier certains axes de réflexion. Enfin, vous y trouverez certains détails quant à nos choix d'implémentation, nous permettant de mettre en valeur et d'exploiter le potentiel de la base de données.

ANALYSE DU PROBLÈME

L'application doit proposer deux vues :

- une vue *client* permettant à l'utilisateur de créer sa simulation et d'obtenir un numéro de dossier, puis de pouvoir consulter le catalogue de l'agence et ajouter à sa simulation des hôtels, des lieux à visiter et des circuits, et enfin confirmer sa simulation. Il pourra alors se présenter à l'agence avec son numéro de dossier et effectuer sa réservation.
- une vue *agence* permettant de vérifier la simulation d'un client venu à l'agence et de la valider en lui ajoutant un paiement. L'agence sera capable de consulter la liste des simulations, des réservations et des clients.

Il sera également vital que de multiples utilisateurs puissent se connecter à la base de données via l'application et modifient la base de données par ajout de simulations, de réservations ou de clients sans conflit.

De plus, afin d'assurer la compatibilité avec les applications partenaires, les relations décrivant les villes, hôtels, circuits et lieux à visiter ne devront pas être modifiées.

L'analyse du problème nous a conduit à l'identification des propriétés élémentaires et à l'élaboration du schéma-entité association correspondant.

CONCEPTION DE LA BASE DE DONNÉES

PROPRIÉTÉS ÉLÉMENTAIRES

{ nomVille, pays, nomLieu, adresseLieu, descriptifLieu, prix, idCircuit, descriptif, villeDépart, paysDepart, villeArrivee, paysArrivee, nbJoursTotal, prixCircuit, dateDepartCircuit, nbPersonnes, ordre, nbJours, nomHotel, adresseHotel, nbChambresTotal, prixChambre, prixPetitDejeuner, numDossier, idClient, typeClient, anneeEnregistrement, nomClient, prenomClient, adresseClient, emailClient, telClient, datePaiement, infoPaiement, nbChambres, dateDepartHotel, dateArriveeHotel, dateVisite, nbPersonnesVisite, nbChambresReservees, nbPetitDejReserves, numDossierReservation }

CONTRAINTES DE VALEUR

$\text{prix} \geq 0$
 $\text{nbJours} \geq 0$
 $\text{nbJoursTotal} \geq 0$
 $\text{prixCircuit} > 0$
 $\text{Ext}(\text{ordre}) \subseteq \llbracket 0, \dots, \text{Max}(\text{Ext}(\text{ordre})) \rrbracket$ $\text{nbChambresTotal} > 0$
 $\text{prixChambre} > 0$

$\text{prixPetitDejeuner} \geq 0$
 $\text{dateDepartHotel} \leq \text{dateArriveeHotel}$
 $\text{nbPersonnesVisite} > 0$
 $\text{nbChambresReservees} > 0$
 $\text{nbPetitDejReserves} \geq 0$
 $\text{Ext}(\text{numDossierReservation}) \subseteq \text{Ext}(\text{numDossier}) \parallel \text{typeClient} \in \text{individuel, societe, groupe}$

DÉPENDANCES FONCTIONNELLES

ISSUES DU SCHÉMA RELATIONEL FOURNI :

$\text{nomLieu, nomVille, pays} \rightarrow \text{adresseLieu, descriptifLieu, prix}$
 $\text{idCircuit} \rightarrow \text{descriptif, villeDepart, paysDepart, villeArrivee, paysArrivee, nbJoursTotal, prixCircuit}$
 $\text{idCircuit, dateDepartCircuit} \rightarrow \text{nbPersonnes}$
 $\text{idCircuit, ordre} \rightarrow \text{nomLieu, nomVille, pays, nbJours}$
 $\text{nomHotel, nomville, pays} \rightarrow \text{adresseHotel, nbChambresTotal, prixChambre, prixPetitDejeuner}$

ISSUES DE NOTRE SGBD :

$\text{idClient} \rightarrow \text{nomClient, prenomClient, typeClient, adresseClient, emailClient, telClient, annéeEnregistrement}$
 $\text{villeDepart, paysDepart} \rightarrow \text{nomVille, pays}$
 $\text{numDossier, idCircuit, dateDepartCircuit} \rightarrow \text{nbPersonnesCircuit}$
 $\text{villeArrivee, paysArrivee} \rightarrow \text{nomVille, pays}$
 $\text{numDossier, nomHotel, nomVille, pays, dateDepartHotel, dateArriveeHotel} \rightarrow \text{nbChambresReservees, nbPetitDejReserves}$
 $\text{numDossier, nomLieu, nomVille, pays, dateVisite} \rightarrow \text{nbPersonnesVisite}$
 $\text{numDossierReservation} \rightarrow \text{datePaiement, infoPaiement}$
 $\text{numDossierReservation} \rightarrow \text{idClient}$

CONTRAINTES DE MULTIPLICITÉ

$\text{numDossier} -| \rightarrow \text{nomHotel, nomVille, pays}$
 $\text{nomHotel, nomVille, pays} -| \rightarrow \text{numDossier}$
 $\text{numDossier} -| \rightarrow \text{nomLieu, nomVille, pays}$
 $\text{nomLieu, nomVille, pays} -| \rightarrow \text{numDossier}$
 $\text{numDossier, dateDepartHotel, dateArriveeHotel} \rightarrow \text{nomHotel, nomVille, pays}$
 $\text{nomHotel, nomVille, pays, dateDepartHotel, dateArriveeHotel} \rightarrow \text{numDossier}$
 $\text{numDossier, dateVisite} \rightarrow \text{nomLieu, nomVille, pays}$
 $\text{nomLieu, nomVille, pays, dateVisite} \rightarrow \text{numDossier}$
 $\text{numDossier} -| \rightarrow \text{idCircuit, dateDepartCircuit}$
 $\text{idCircuit, dateDepartCircuit} -| \rightarrow \text{numDossier}$
 $\text{idClient} \rightarrow \text{numDossierReservation}$
 $\text{idCircuit} \rightarrow \text{idCircuit, dateDepartCircuit}$
 $\text{nomVille, pays} -| \rightarrow \text{nomLieu, nomVille, pays}$
 $\text{nomVille, pays} -| \rightarrow \text{nomHotel, nomVille, pays}$
 $\text{nomLieu, nomVille, pays} -| \rightarrow \text{idCircuit, ordre}$
 $\text{nomVille, pays} -| \rightarrow \text{villeDepart, paysDepart}$
 $\text{nomVille, pays} -| \rightarrow \text{villeArrivee, paysArrivee}$

AUTRES CONTRAINTES

“Un dossier contient au moins une réservation (hôtel, circuit, lieux).”
 “Il ne peut y avoir deux villes de même nom dans le même pays.”
 “Un circuit commence par une étape de départ et se termine par une étape d’arrivée. Tout circuit contient au moins deux étapes : le départ et l’arrivée.”
 “Un client doit avoir au moins une réservation (simulation payée).”
 “Au paiement, et pour permettre la réservation du voyage, il faut vérifier la disponibilité des hôtels et circuits réservés (nombre de places libres compatible avec nombre de places réservées) ainsi que toutes les contraintes.”

SCHÉMA ENTITÉ-ASSOCIATION

PASSAGE AU RELATIONNEL

TYPES D'ENTITÉ SIMPLE

Ville (nomVille, pays)

Circuit (idCircuit, descriptif, villeDepart, paysDepart, villeArrivee, paysArrivee, nbJoursTotal, prixCircuit)

Client (idClient, nomClient, prenomClient, typeClient, adresseClient, emailClient, telClient, anneeEnregistrement)

Simulation (numDossier)

TYPES D'ENTITÉ FAIBLE

ENTITÉS FAIBLES DE VILLE :

Hotel (nomHotel, ville, pays, adresseHotel, nbChambresTotal, prixChambre, prixPetitDejeuner)

LieuxAvisiter (nomLieu, ville, pays, adresseLieu, descriptifLieu, prix)

ENTITÉS FAIBLES DE CIRCUIT :

DateCircuit (idCircuit, dateDepartCircuit, nbPersonnes)

Etales (idCircuit, ordre, nomLieu, ville, pays, nbJours)

SOUS-TYPES D'ENTITÉ

Reservation(numDossier, datePaiement, infoPaiement, idClient)

L'entité Reservation est un sous-type d'entité de Simulation. Nous avons décidé d'opter pour la duplication de la clé numDossier de Simulation dans l'entité Reservation afin de pouvoir facilement différencier les simulations des réservations et afin de pouvoir traduire la relation avec le Client. Ce choix a été renforcé par le faible poids mémoire de l'attribut numDossier et par l'absence de modification d'une réservation après son paiement.

ASSOCIATIONS

Les entités liées à la réservations d'hôtels, de lieux à visiter et de circuits sont issus d'associations entre l'entité Client et les entités Hotel, LieuxAvisiter et Circuit.

ReserveHotel (nomHotel, ville, pays, numDossier, dateDepartHotel, dateArriveeHotel, nbChambresReservees, nb-PetitDejReserves)

ReserveCircuit (idCircuit, dateDepartCircuit, numDossier, nbPersonnesCircuit)

ReserveVisite (nomLieu, ville, pays, numDossier, dateVisite, nbPersonnesVisite)

FORMES NORMALES

Toutes les relations sont *3FN* car pour chacune tous les attributs non clé sont pleinement dépendant et directement issus des clés de l'entité à laquelle elles appartiennent. //TODO 3FNBCK il me semble plutôt

ANALYSE DES FONCTIONNALITÉS

Les transactions sont sérialisables afin d'isoler chaque utilisateur et permettre des transactions simultanées sur la base de données sans en perdre la cohérence.

FONCTIONNALITÉS LIÉES AU CLIENT :

- consulter le catalogue : sélection de la table demandée ;
- créer une simulation : insertion dans la table *Simulation* ;

- obtenir un numéro de dossier : faire une requête à la base de donnée pour obtenir ce numéro ;
- ajouter des réservations à sa simulation : insertion dans les tables correspondante ; *ReserveHotel*, *ReserveVisite* et *ReserveCircuit*.

FONCTIONNALITÉS LIÉES À L'AGENCE :

- voir les clients, simulations, réservations : sélection des tables correspondantes ;
- ajouter un nouveau client : insertion dans la table *Client* et obtention d'un identifiant client via la base de donnée ;
- lier une réservation à un client existant : recherche par sélection dans la table *Client* ;
- confirmer une simulation, ajouter un paiement à une réservation : insertion dans la table *Reservation* ;
- tester les disponibilités d'une simulation : vérification de la compatibilité entre le nombre de places disponibles et le nombre de places demandées, sur les périodes données pour les hôtels et aux dates données pour les circuits ;
- obtenir un récapitulatif d'une simulation : sélectionner les réservations via un numéro de dossier et calculer le nombre de personnes total, les dates de voyage et le prix total.

Les détails de l'implémentation de ces transactions sont disponibles dans le script ... //TODO

BILAN DU PROJET

ORGANISATION

Semaine 1	Présentation et analyse du problème
Semaine 2	Analyse fonctionnelle
Semaine 3	Schéma entité-association Réunion de suivi Révision des dépendances et du schéma entité-association suite au suivi (version finale)
Semaine 4	SQL : tables, transactions Application Java : connexion, dialogue, requêtes, visualisation des tables (catalogue) Script de peuplement
Semaine 5	Application Java : sérialisation des transactions, fonctionnalités Documentation
Semaine 6	Soutenance

Les semaines 1 à 3 ont été dédiées à définir et réfléchir de manière collective aux fondations de notre base de données. Le script de création de la base de données a également été réalisé en groupe. A partir de la semaine 4, le développement s'est effectué en parallèle. Chaque personne était dédiée à une tâche entre les scripts de peuplement de la base de donnée, l'implémentation des transactions et le développement de l'application. La base de données étant bien en place la semaine 5, toute l'équipe s'est alors concentrée sur l'application afin de réaliser toutes les fonctionnalités identifiées. A l'issue de cette semaine, la documentation a été rédigée et la soutenance préparée.

Afin d'organiser notre travail, nous avons utilisé des outils de travail collaboratif de type *Google Docs* et *draw.io* afin de réaliser l'analyse fonctionnelle et le schéma entité-association. En ce qui concerne le code (script et application), nous avons utilisé le système de tickets de *Github* afin de garder une trace de notre avancement et des problèmes rencontrés et de répartir les tâches de manière efficace entre les membres de notre équipe.

DIFFICULTÉS RENCONTRÉES

ANALYSE DU SUJET, ATTENTES DU CLIENT :

La difficulté majeure à laquelle nous avons fait face est l'analyse du sujet. Malgré de nombreuses questions posées à l'expert technique et au client ainsi qu'aux autres équipes sur le projet, nous avons continué tout au long du projet à nous poser des questions relatives aux attentes de l'agence AWHY vis-à-vis de notre produit. Nous avons notamment eu des difficultés à propos des relations entre une simulation, une réservation et un client. Cela a conduit à revenir à plusieurs reprises sur le schéma entité-association et l'analyse fonctionnelle ainsi que de nombreux débats

au sein de l'équipe pour arriver à une entente commune.

GÉNÉRATION D'IDENTIFIANT UNIQUE :

La génération d'identifiant unique pour les dossiers de simulation et les identifiants client est une difficulté que nous avons rapidement identifiée. Finalement, nous avons laissé la base de donnée la tâche de générer ces identifiants à l'aide de séquences SQL. La création d'un client ou d'un dossier commence donc par une requête à la base de donnée afin de demander un identifiant.

VÉRIFICATION D'UNE SIMULATION :

La vérification d'une simulation a occupé plusieurs membres de notre équipe. Plus qu'une difficulté, cela a été une tâche conséquente car nous devons effectuer de nombreuses requêtes afin de récupérer diverses informations telles que le prix total, le nombre de places disponibles ou encore les dates de voyage.

DIALOGUE INTUITIF AVEC L'UTILISATEUR :

Nous avons rencontré à plusieurs reprises des difficultés à créer une application claire et intuitive pour l'utilisateur. Bien que cela ne soit pas le point principal de ce projet, nous voulions pouvoir présenter à notre client une application attrayante. Nous avons opté pour l'utilisation de multiples popup avec peu d'informations. Par exemple en ce qui concerne la réservation de circuit, nous commençons par montrer au client les étapes dans une popup. Il peut choisir sa date et indiquer le nombres de personnes effectuant le circuit via une nouvelle popup. Puis, pour chaque étape, nous lui proposons de réserver un hôtel.

RESSENTI FINAL DE L'ÉQUIPE

Ce projet est le premier concernant la conception d'une base de données. De ce fait, nous nous sommes sentis dans un premier temps un peu perdus quant à l'analyse fonctionnelle et le schéma entité-association. C'est pourquoi nous avons passé beaucoup de temps à discuter ensemble des fondations de notre base de données. Nous pensons qu'il était vital de pouvoir se reposer sur des fondations solides afin de ne pas avoir à revenir sur la conception de la base de données lors du développement de l'application.

Globalement, nous sommes satisfaits de notre rendu. Avec plus de temps, nous souhaiterions mettre au clair certains points avec le client. Cela pourrait potentiellement conduire à l'altération de notre base de données mais permettrait de mieux coller aux attentes de notre client. De plus, nous souhaiterions ajouter de nouvelles fonctionnalités à l'application telles que le filtrage du catalogue ou une meilleure utilisation du système de "panneaux" lors d'une simulation. Enfin, nous souhaiterions augmenter l'ergonomie de notre application, par exemple en pré-remplissant certains champs de dates ou de nombres de personnes.